**Walchand College of Engineering, Sangli**

**Department of Computer Science and Engineering**

**Semester: 1**

**Class: Final Year (Computer Science and Engineering)**

**Year: 2022-23**

Course: High Performance Computing Lab

Name: Sakshi Sanjay Desai

Prn No.2019BTECS00021

Batch:B1

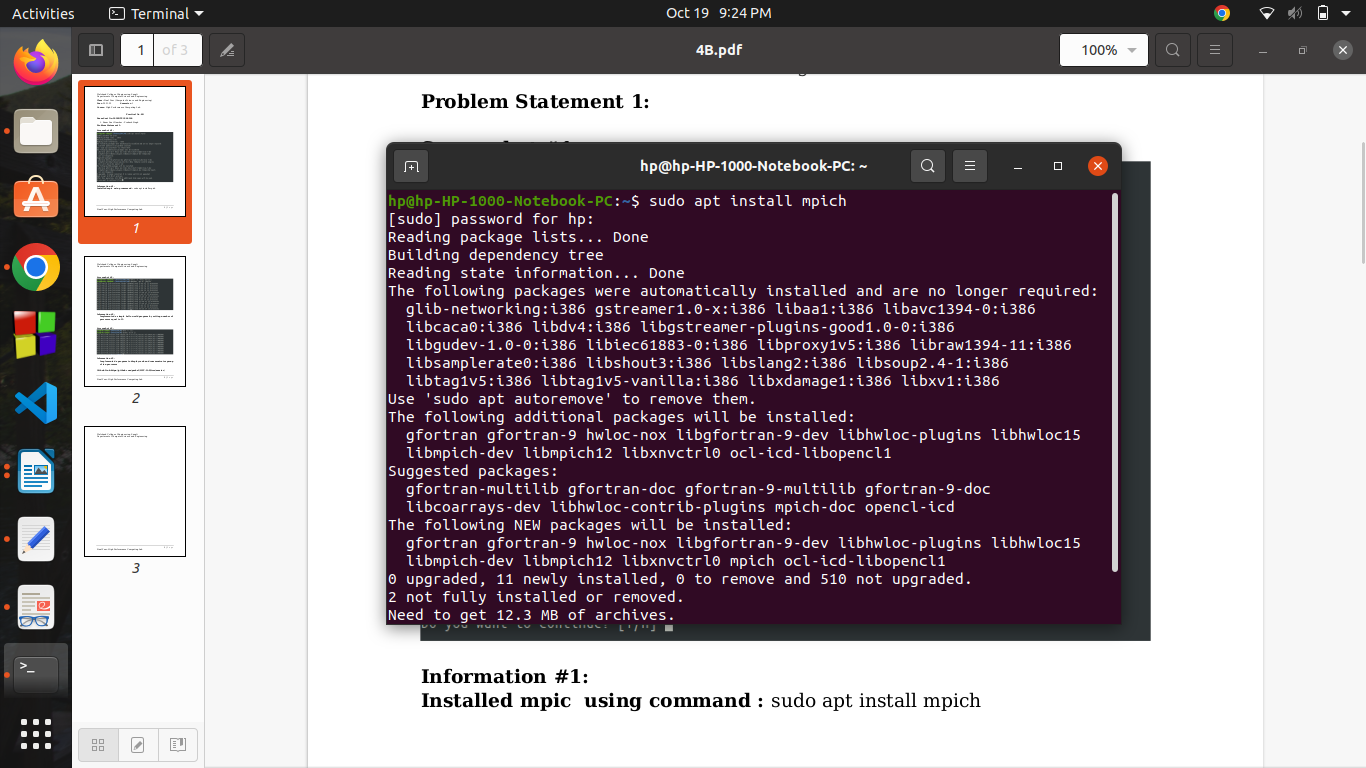
**Practical No. 5**

**Installation of MPI and implementation of basic functions of MPI**

Complete the installation of MPI on the platform chosen by you.

Installing MPI in local machine

Installed using command : sudo apt install mpich



**Q1: Implement a simple hello world program by setting number of processes equal to 10.**

code:

#include <mpi.h>

#include <stdio.h>

int main( int argc, char \*argv[] )

{

int rank, size;

MPI\_Init( &argc, &argv );

MPI\_Comm\_rank( MPI\_COMM\_WORLD, &rank );

MPI\_Comm\_size( MPI\_COMM\_WORLD, &size );

char processor\_name[MPI\_MAX\_PROCESSOR\_NAME];

int name\_len;

MPI\_Get\_processor\_name(processor\_name,&name\_len);

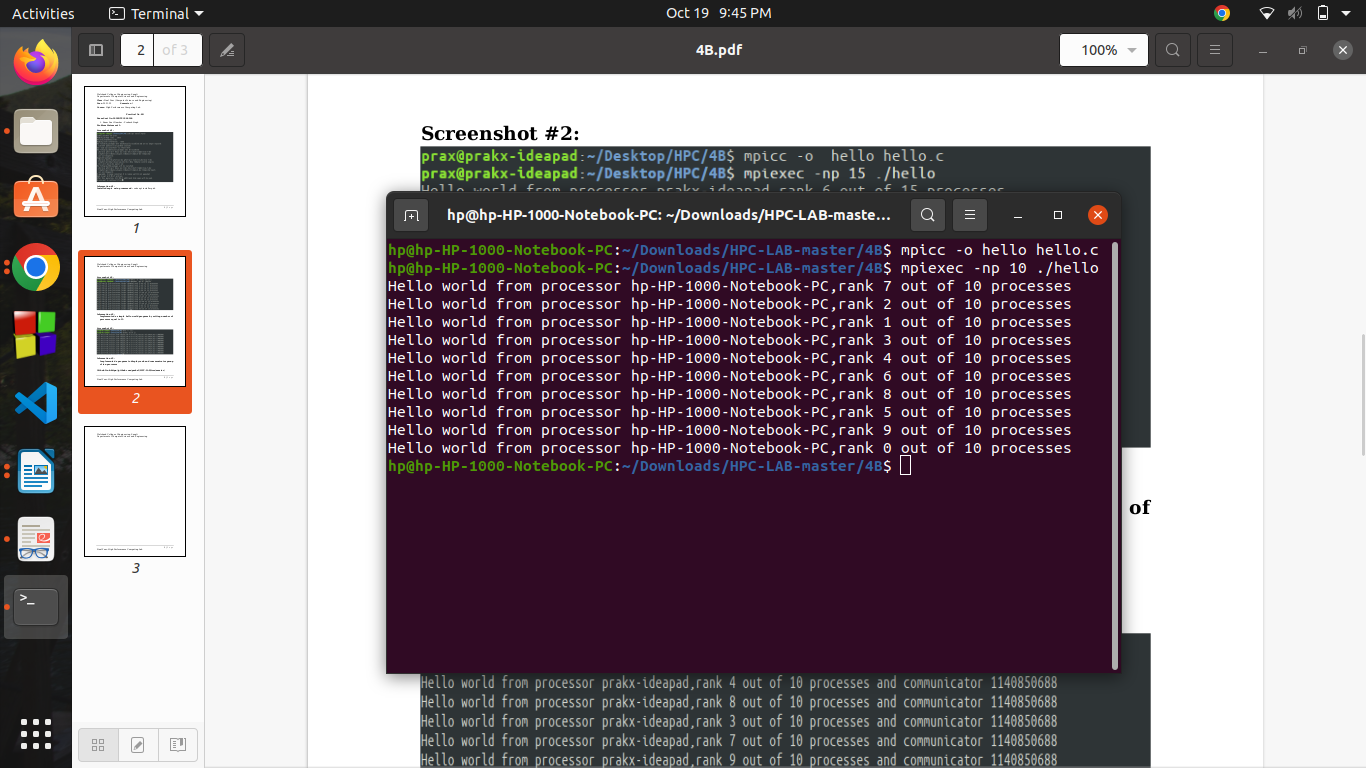
printf( "Hello world from processor %s,rank %d out of %d processes \n",processor\_name, rank, size );

MPI\_Finalize();

return 0;

}

output:



**Q2. Implement a program to display rank and communicator group of five processes.**

Code:

#include <mpi.h>

#include <stdio.h>

int main( int argc, char \*argv[] )

{

int rank, size;

MPI\_Init( &argc, &argv );

MPI\_Comm\_rank( MPI\_COMM\_WORLD, &rank );

MPI\_Comm\_size( MPI\_COMM\_WORLD, &size );

char processor\_name[MPI\_MAX\_PROCESSOR\_NAME];

int name\_len;

MPI\_Get\_processor\_name(processor\_name,&name\_len);

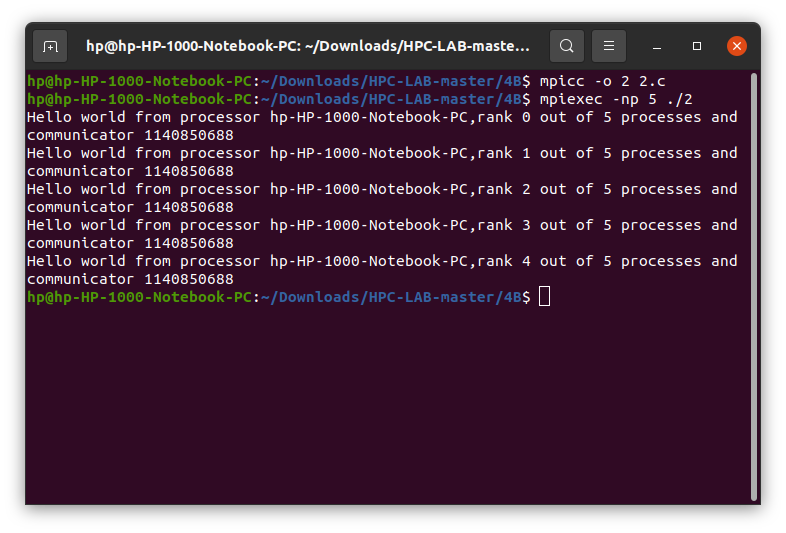
printf( "Hello world from processor %s,rank %d out of %d processes and communicator %d \n",processor\_name, rank, size,MPI\_COMM\_WORLD);

MPI\_Finalize();

return 0;

}

output:

****